PCT

NOTIFICATION OF ELECTION

(PCT Rule 61.2)

From the INTERNATIONAL BUREAU

To:

Commissioner
US Department of Commerce
United States Patent and Trademark
Office, PCT
2011 South Clark Place Room
CP2/5C24
Arlington, VA 22202
ETATS-UNIS D'AMERIQUE

in its capacity as elected Office

Date of mailing (day/month/year) 10 January 2001 (10.01.01)	
International application No. PCT/GB00/00783	**Applicant's or agent's file reference** A25765 WO
International filing date (day/month/year) 03 March 2000 (03.03.00)	**Priority date** (day/month/year) 15 March 1999 (15.03.99)
Applicant WADDINGTON, Daniel, Giles	

1. The designated Office is hereby notified of its election made:

[X] in the demand filed with the International Preliminary Examining Authority on:

27 September 2000 (27.09.00)

[ ] in a notice effecting later election filed with the International Bureau on:

2. The election [X] was

[ ] was not

made before the expiration of 19 months from the priority date or, where Rule 32 applies, within the time limit under Rule 32.2(b).

The International Bureau of WIPO **34, chemin des Colombettes** **1211 Geneva 20, Switzerland**	Authorized officer Jean-Marc Vivet
Facsimile No.: (41-22) 740.14.35	Telephone No.: (41-22) 338.83.38

PATENT COOPERATION TREATY

PCT

INTERNATIONAL SEARCH REPORT

(PCT Article 18 and Rules 43 and 44)

Applicant's or agent's file reference	FOR FURTHER ACTION	see Notification of Transmittal of International Search Report (Form PCT/ISA/220) as well as, where applicable, item 5 below.
A25765 WO		
Int rnational application No.	International filing date *(day/month/year)*	(Earliest) Priority Date *(day/month/year)*
PCT/GB 00/ 00783	03/03/2000	15/03/1999

Applicant

BRITISH TELECOMMUNICATIONS PUBLIC LIMITED COMPANY

This International Search Report has been prepared by this International Searching Authority and is transmitted to the applicant according to Article 18. A copy is being transmitted to the International Bureau.

This International Search Report consists of a total of 3 sheets.

[X] It is also accompanied by a copy of each prior art document cited in this report.

1. **Basis of the report**

   a. With regard to the **language,** the international search was carried out on the basis of the international application in the language in which it was filed, unless otherwise indicated under this item.

   [ ] the international search was carried out on the basis of a translation of the international application furnished to this Authority (Rule 23.1(b)).

   b. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, the international search was carried out on the basis of the sequence listing :

   [ ] contained in the international application in written form.

   [ ] filed together with the international application in computer readable form.

   [ ] furnished subsequently to this Authority in written form.

   [ ] furnished subsequently to this Authority in computer readble form.

   [ ] the statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.

   [ ] the statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished

2. [ ] **Certain claims were found unsearchable** (See Box I).

3. [ ] **Unity of invention is lacking** (see Box II).

4. With regard to the **title,**

   [X] the text is approved as submitted by the applicant.

   [ ] the text has been established by this Authority to read as follows:

5. With regard to the **abstract,**

   [X] the text is approved as submitted by the applicant.

   [ ] the text has been established, according to Rule 38.2(b), by this Authority as it appears in Box III. The applicant may, within one month from the date of mailing of this int rnational search report, submit comments to this Authority.

6. The figure of the **drawings** to be published with the abstract is Figure No. 2

   [ ] as suggested by the applicant.

   [X] because the applicant failed to suggest a figure.

   [ ] because this figure better characterizes th invention.

   [ ] None of the figures.

INTERNATIONAL SEARCH REPORT

International Application No
P[redacted]B 00/00783

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F9/48 G06F9/50

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 0 798 638 A (HITACHI LTD) 1 October 1997 (1997-10-01) column 8, line 22 -column 14, line 20 column 32, line 56 -column 33, line 24 ---	1-6,8,9, 11,12, 15,16
A	US 5 812 844 A (DRAVES JR RICHARD P ET AL) 22 September 1998 (1998-09-22) column 6, line 56 -column 10, line 23 claims 1-3 ---	1-9,11, 12,15,16
A	EP 0 817 041 A (SUN MICROSYSTEMS INC) 7 January 1998 (1998-01-07) column 2, line 54 -column 5, line 31 claims 1-7 --- -/--	1-4

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the International search	Date of mailing of the international search report
7 September 2000	13/09/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Bijn, K

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/GB 00/00783

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0798638 A	01-10-1997	CA 2200929 A	28-09-1997
		JP 2904483 B	14-06-1999
		JP 9319597 A	12-12-1997
		US 5944778 A	31-08-1999
US 5812844 A	22-09-1998	NONE	
EP 0817041 A	07-01-1998	US 5826082 A	20-10-1998
		JP 10063519 A	06-03-1998
US 5247677 A	21-09-1993	FR 2691557 A	26-11-1993
		JP 6035726 A	10-02-1994
EP 0790557 A	20-08-1997	JP 9282184 A	31-10-1997
EP 0658841 A	21-06-1995	US 5487170 A	23-01-1996
		JP 7200318 A	04-08-1995

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 99 30 1952

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X A	EP 0 798 638 A (HITACHI LTD) 1 October 1997 (1997-10-01) * column 8, line 22 - column 14, line 20 * * column 32, line 56 - column 33, line 24 * ---	1,4,6,7, 9,15 2,3,5, 10-12	G06F9/46
X A	US 5 812 844 A (DRAVES JR RICHARD P ET AL) 22 September 1998 (1998-09-22) * column 6, line 56 - column 10, line 23 * * claims 1-3 * ---	1,3,4,15 2,5-12	
X A	EP 0 817 041 A (SUN MICROSYSTEMS INC) 7 January 1998 (1998-01-07) * column 2, line 54 - column 5, line 31 * * claims 1-7 * ---	1,3 2,4,5	
A	US 5 247 677 A (WELLAND ROBERT V ET AL) 21 September 1993 (1993-09-21) * column 5, line 24 - column 6, line 53 * ---	1,13	TECHNICAL FIELDS SEARCHED (Int.Cl.7)
A	EP 0 790 557 A (MATSUSHITA ELECTRIC IND CO LTD) 20 August 1997 (1997-08-20) * page 8, line 22 - page 13, line 38 * ---	1,13	G06F
A	EP 0 658 841 A (IBM) 21 June 1995 (1995-06-21) * column 5, line 34 - column 7, line 8 * -----	1,10	

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	29 September 1999	Bijn, K

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

1

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 99 30 1952

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

29-09-1999

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0798638 A	01-10-1997	CA 2200929 A	28-09-1997
		JP 2904483 B	14-06-1999
		JP 9319597 A	12-12-1997
US 5812844 A	22-09-1998	NONE	
EP 0817041 A	07-01-1998	US 5826082 A	20-10-1998
		JP 10063519 A	06-03-1998
US 5247677 A	21-09-1993	FR 2691557 A	26-11-1993
		JP 6035726 A	10-02-1994
EP 0790557 A	20-08-1997	JP 9282184 A	31-10-1997
EP 0658841 A	21-06-1995	US 5487170 A	23-01-1996
		JP 7200318 A	04-08-1995

14

TENT COOPERATION TR Y

PCT

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

(PCT Article 36 and Rule 70)

Applicant's or agent's file reference	FOR FURTHER ACTION	See Notification of Transmittal of International Preliminary Examination Report (Form PCT/IPEA/416)
A25765 WO		
International application No.	International filing date *(day/month/year)*	Priority date *(day/month/year)*
PCT/GB00/00783	03/03/2000	15/03/1999
International Patent Classification (IPC) or national classification and IPC G06F9/46		
Applicant BRITISH TELECOMMUNICATIONS P.L.C.et al.		

1. This international preliminary examination report has been prepared by this International Preliminary Examining Authority and is transmitted to the applicant according to Article 36.

2. This REPORT consists of a total of 8 sheets, including this cover sheet.

   ☐ This report is also accompanied by ANNEXES, i.e. sheets of the description, claims and/or drawings which have been amended and are the basis for this report and/or sheets containing rectifications made before this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions under the PCT).

   These annexes consist of a total of sheets.

3. This report contains indications relating to the following items:

   - I ☒ Basis of the report
   - II ☐ Priority
   - III ☐ Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
   - IV ☒ Lack of unity of invention
   - V ☒ Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations suporting such statement
   - VI ☐ Certain documents cited
   - VII ☒ Certain defects in the international application
   - VIII ☒ Certain observations on the international application

Date of submission of the demand	Date of completion of this report
27/09/2000	11.06.2001
Name and mailing address of the international preliminary examining authority: European Patent Office D-80298 Munich Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	Authorized officer Thibaudeau, J Telephone No. +49 89 2399 2349

Form PCT/IPEA/409 (cover sheet) (January 1994)

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No. PCT/GB00/00783

I. Basis of the r port

1. With regard to the **elements** of the international application *(Replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to this report since they do not contain amendments (Rules 70.16 and 70.17))*:
Description, pages:

1-13 as originally filed

Claims, No.:

1-16 as originally filed

Drawings, sheets:

1-5 as originally filed

2. With regard to the **language**, all the elements marked above were available or furnished to this Authority in the language in which the international application was filed, unless otherwise indicated under this item.

These elements were available or furnished to this Authority in the following language: , which is:

- ☐ the language of a translation furnished for the purposes of the international search (under Rule 23.1(b)).
- ☐ the language of publication of the international application (under Rule 48.3(b)).
- ☐ the language of a translation furnished for the purposes of international preliminary examination (under Rule 55.2 and/or 55.3).

3. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, the international preliminary examination was carried out on the basis of the sequence listing:

- ☐ contained in the international application in written form.
- ☐ filed together with the international application in computer readable form.
- ☐ furnished subsequently to this Authority in written form.
- ☐ furnished subsequently to this Authority in computer readable form.
- ☐ The statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.
- ☐ The statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished.

4. The amendments have resulted in the cancellation of:

- ☐ the description, pages:
- ☐ the claims, Nos.:

☐ the drawings, sheets:

5. ☐ This report has been established as if (some of) the amendments had not been made, since they have been considered to go beyond the disclosure as filed (Rule 70.2(c)):

   (Any replacement sheet containing such amendments must be referred to under item 1 and annexed to this report.)

6. Additional observations, if necessary:

IV. Lack of unity of invention

1. In response to the invitation to restrict or pay additional fees the applicant has:

   ☐ restricted the claims.

   ☐ paid additional fees.

   ☐ paid additional fees under protest.

   ☒ neither restricted nor paid additional fees.

2. ☐ This Authority found that the requirement of unity of invention is not complied and chose, according to Rule 68.1, not to invite the applicant to restrict or pay additional fees.

3. This Authority considers that the requirement of unity of invention in accordance with Rules 13.1, 13.2 and 13.3 is

   ☐ complied with.

   ☒ not complied with for the following reasons:
   see separate sheet

4. Consequently, the following parts of the international application were the subject of international preliminary examination in establishing this report:

   ☐ all parts.

   ☒ the parts relating to claims Nos. 1-4, 15, 16.

V. Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Yes:	Claims	1-4,15,16
	No:	Claims	
Inventive step (IS)	Yes:	Claims	
	No:	Claims	1-4,15,16

Industrial applicability (IA)	Yes:	Claims	1-4,15,16
	No:	Claims	

2. Citations and explanations
see separate sheet

VII. Certain defects in the international application

The following defects in the form or contents of the international application have been noted:
see separate sheet

VIII. Certain observations on the international application

The following observations on the clarity of the claims, description, and drawings or on the question whether the claims are fully supported by the description, are made:
see separate sheet

1. Reference is made to the following document:

   D1 = EP-A-0 798 638 (HITACHI LTD) 1 October 1997 (1997-10-01)

2. **Item IV: Lack of unity of invention**

   The separate groups of invention are:

2.1 A method of administering resource utilisation in a computer, said method running a first process to make reservation for access to a resource and a second process to grant requests for access to said resource, said method further creating a scheduling means for processing reservation requests and a reservation means for making reservations for access to a resource (claim 1);

2.2 A method of administering resource utilisation in a computer, said method running a first process to make reservation for access to a resource and a second process to grant requests for access to said resource, said first process allocating resource tokens and said second process generating a resource token identifier (claim 13);

2.3 A method of administering resource utilisation in a computer, said method running a first process to make reservation for access to a resource and a second process to grant requests for access to said resource, said first process determining a weighting function and said second process using a stochastic process (claim 14);

2.4 A method of scheduling or granting access to a CPU using a one-dimensional reservation request pattern to be merged with a one-dimensional CPU access control pattern representing empty CPU access time slots (claims 5 and 9).

The common concept linking 2.1, 2.2, 2.3 and 2.4 is merely a method of administering resource utilisation in a computer, said method making a reservation for access to a resource (claim 5 and first process of claims 1, 13 and 14) and granting requests for access to said resource (claim 9 and second process of claims 1, 13 and 14).

However D1 discloses such a method, the resource in question being CPU time allocated to a process group (see column 8, lines 23-34).

Therefore said common concept is not new (Article 33(2) PCT).

Thus the separate groups of invention mentioned above are not so linked as to form a single general inventive concept (Rule 13.1 PCT).

The applicant neither restricted the claims nor paid additional fees. Therefore the international preliminary examination report has been established on claims 1-4, 15 and 16 which appear to relate to the main invention.

3. **Item V: Reasoned statement under Rule 66.2(a)(ii) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement**

As to claim 1:
D1 discloses a method of administering resource utilisation in a computer (see Figure 1), the method comprising the steps of:
- running a first process (periodic kernel process 101) to make a reservation for access to a resource (CPU time) in dependence on a resource requirement communication from an application process (group master 102);
- running a second process (periodic kernel process 101) to grant requests for access to said resource from said application process in dependence on said reservation (see column 8, lines 36-44);
- creating a scheduling means and a reservation means (scheduling table 900 in association with a scheduler (101)) having a method ("alloc_time_slot" function - see column 8, lines 30-34) for processing reservation requests for a plurality of resources and for making reservations for access to a resource (see column 8, lines 36-38).
D1 further discloses that said application process (group master 102) calls a method of the scheduling means and a method of the reservation means ("alloc_time_slot" function - see column 8, lines 23-27) to make a reservation.
D1 also discloses the steps of:
- running a resource specific (CPU) scheduling process to grant access to a resource in dependence on the reservation made by the reservation means; and
- utilising said resource for the purposes of said application process.

D1 does not disclose the calling of two different methods, one for processing the reservation requests, the other for making reservations. However the use of two different methods or only one ("alloc_time_slot" function) to perform the same functions is merely a matter of definition. The fact that scheduling and reservation are made by two different means, whereas D1 uses the same means (scheduling table

900 in association with a scheduler (101)) is just a matter of design option.

D1 does not disclose two different resource access requirement definitions as parameter for the scheduling method and the reservation method. However said different definitions are not detailed in the claims so that no inventive step can be seen in merely defining a first and a second resource access requirement definitions, which could also be identical.

As to claim 2:
Since the resource access requirement definitions are not detailed, no inventive step can be seen in stating that a first one is translated into a second one.

As to claims 3 and 4:
D1 discloses the CPU as a resource (see comments above). D1 also discloses a mass storage device as a resource (see Figure 3).

As to claims 15 and 16:
Since they refer to claims 1-4, their subject-matter also lacks inventive step.

4. **Item VII: Certain defects in the international application**

i) To meet the requirements of Rule 6.3(b) PCT the independent claims should have been properly cast in the two part form, with those features which in combination are part of the prior art (see document D1) being placed in the preamble.

ii)The features of the claims are not provided with reference signs placed in parentheses (Rule 6.2(b) PCT).

iii) The description is not in conformity with the claims as required by Rule 5.1(a)(iii) PCT.

5. **Item VIII: Certain observations on the international application**

The application does not meet the requirements of Article 6 PCT, because claim 1 is not clear.

Claim 1 comprises a first process run to make a reservation for access to a resource in dependence on a resource requirement communication from an application.

However claim 1 further comprises the creation of scheduling means having methods for processing reservation requests for a plurality of resources and initiating resource specific reservation processing and the creation of reservation means having methods for making reservations for access to a resource.

It is therefore not clear whether said scheduling means and said reservation means, or at least their creation, are part of the first process.

Indeed the application process calls a method of the scheduling means which, in turn, calls a reservation method of the reservation means to make a reservation for said application process whereas said first process is supposed to make the reservation.

The same objection applies for the second process. Indeed claim 1 comprises a second process run to grant requests for access to the resource from the application process in dependence on the reservation. However claim 1 also comprises a resource specific scheduling process run to grant access to a resource in dependence on the reservation. It is therefore not clear in claim 1 whether said second process and said resource specific scheduling process correspond to the same process entity.

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 :
G06F 9/46

A2

(11) International Publication Number: WO 00/55732

(43) International Publication Date: 21 September 2000 (21.09.00)

(21) International Application Number: PCT/GB00/00783

(22) International Filing Date: 3 March 2000 (03.03.00)

(30) Priority Data:

99301952.0	15 March 1999 (15.03.99)	EP
99303936.1	20 May 1999 (20.05.99)	EP

(71) Applicant *(for all designated States except US)*: BRITISH TELECOMMUNICATIONS PUBLIC LIMITED COMPANY [GB/GB]; 81 Newgate Street, London, EC1A 7AJ (GB).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: WADDINGTON, Daniel, Giles [GB/GB]; Flat 1, 25 St. Georges Quay, Lancaster, Lancashire LA1 4YR (GB).

(74) Agent: WILSON, Peter, David; BT Group Legal Services, Intellectual Property Department, Holborn Centre, 8th floor, 120 Holborn, London EC1N 2TE (GB).

(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published

Without international search report and to be republished upon receipt of that report.

(54) Title: RESOURCE SCHEDULING

(57) Abstract

A method of administering resource utilisation in a computer comprises running a first process (13, 14, 15) to make a reservation for access to a resource in dependence on a resource requirement communication from an application (21) and running a second process (16, 17, 19) to grant requests for access to said resource from said application in dependence on said reservation. A further process (12) provides a common interface between the first process (13, 14, 15) for each resource and the application (21). The further process (12) converts high–level abstract resource requirement definitions into formats applicable to the first process (13, 14, 15) for the resource in question. The processes are preferably implemented at methods of software objects.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

RESOURCE SCHEDULING

The present invention relates to resource scheduling in a computer.

Enterprise level general-purpose operating systems, both workstation and server, are traditionally designed for efficient resource utilisation. Applications share a common set of resources, which are initially scheduled according to their priority and then in a fair weighted manner. This approach results is both an efficient and fair division of resources to user-level applications. However, certain applications, in particularly multimedia processing applications, are inherently sensitive to resource availability. As a result, the use of priority-based resource federation policies often results in such applications failing to meet their processing goals.

US 5,812,844 discloses a method of CPU scheduling, using the Earliest Deadline First scheduling of threads. However, there is no notion of resource admission in this invention and therefore guarantees are only best-effort and the scheduling is only single level, based on threads. Scheduling dependent upon process priority and/or urgency is not included. EP-A-0 798 638 discloses a further approach to CPU scheduling, in which a number of processes exist as a group. In turn the group is 'scheduled' by increasing its priority, although the scheduler cannot guarantee that a process within a group will be scheduled.

Conventionally, relatively complex processing is required of the scheduling process at the stage where a process is granted access to a resource. The present invention enables a reduction in the complexity of the processing at the access grant stage by providing for reservation of resources in advance.

According to the present invention, there is provided a method of administering resource utilisation in a computer, the method comprising the steps of: running a first process to make a reservation for access to a resource in dependence on a resource requirement communication from an application process; running a second process to grant requests for access to said resource from said application process in dependence on said reservation; creating a scheduling means having a method or methods for processing reservation requests for a plurality of resources and initiating resource specific reservation processing; creating a reservation means having a method or methods for making reservations for access to a resource; said application process calling a method of the

scheduling means, said method taking a first resource access requirement definition as a parameter; said method of the scheduling means calling a reservation method of the reservation means to make a reservation for said application process, the reservation method taking a second resource access requirement definition as a parameter; running a resource specific scheduling process to grant access to a resource in dependence on the reservation made by the reservation means; and utilising said resource for the purposes of said application process. The reservation request may be made by an application *per se* or a sub-process or component thereof. Furthermore, access may be granted to any process of an application or for a particular sub-unit of the application for which the reservation request was made.

Making reservations and granting access at application level assists in embodiments of the present invention intended to operate with legacy code. However, where all applications are written for an operating system employing a scheduling method according to the present invention, component or thread level reservation and access grant is preferred.

Preferably, said method of the scheduling means translates the first resource requirement definition into the second resource requirement definition.. The advantage of this is that the resource request can be defined by a programmer in a hardware independent manner and the scheduling object for a particular platform can translate that programmer-produced resource request in a potentially non-linear manner on the basis of the properties of the particular platform.

A method according to the present invention can be applied to the allocation of CPU time, access to mass storage devices, e.g. hard disk drives, optical disk drives and the like, and memory management.

Preferably in the case of mass storage access, said first process comprises allocating one or more "lottery ticket" numbers to said application or a process thereof in dependence on the second resource access requirement definition and storing requests for access to a mass storage device from application processes; generating substantially randomly a lottery ticket number; and if no application process has been allocated said lottery ticket number then passing on to a mass storage device driver process the stored request for access from an application process selected on the basis of a predetermined prioritisation criterion else

passing on to a mass storage device driver process a stored request for access from an application process to which said lottery ticket number was allocated.

The majority of multi-tasking operating systems, including Windows NT, Unix and Linux, use a scheduling algorithm known as Round-Robin. At the end of a time slot, or when a thread has revoked its time slice, the dispatcher takes off the first thread which is waiting on the highest priority queue (some queues will be empty). Additional techniques are used to prevent starvation by dynamically increasing the priority of long-waiting threads.

According to the present invention, there is also provided a method of scheduling access to a CPU which may be implemented in a method of administering resource utilisation in a computer according to the present invention, the method comprising the steps of: generating a one-dimensional reservation request pattern; merging the reservation request pattern with a one-dimensional CPU access control pattern, representing empty CPU access time slots and reserved CPU access time slots, without substantially disturbing either the reservation request pattern or the reserved CPU access time slots in the reservation request pattern.

The reservation request pattern may be shorter than the CPU access control pattern, in which case the reservation request pattern is effectively extended by repetition in the merging process. Preferably, said merging step comprises relocating a non-empty time slot element of the reservation request pattern or the CPU access control pattern such that the patterns can be merged without any reserved CPU access time slot elements being deleted or overwritten. More preferably, the relocated non-empty time slot element is relocated by an amount defined in said time slot element. Both forwards and backwards shifts or shift limits may be included in each element.

According to the present invention, there is provided a method of granting access to a CPU which may be implemented in a method of administering resource utilisation in a computer according to the present invention, the method comprising:

generating a one-dimensional CPU access control pattern, each element of which relates to a quantum of CPU access time; and at the end of a quantum of CPU access time:

granting access to any pending processes having a priority greater than a predetermined level; and then if the next pattern element is empty then granting access to a pending process meeting a predetermined prioritisation (e.g. Round-Robin) criterion else granting access to a process identified in the pattern element.

Preferably, an entry for the process in any of a plurality of different priority queues will be searched for and access will be granted in respect of the entry for the process having the highest priority. Alternatively, access is granted to the process identified in the pattern element when there is not a populated process queue having a higher priority than the queue in which said process is present. Preferably, the a one-dimensional CPU access control pattern is generated by a method of scheduling access to a CPU according to the present invention.

Embodiments of the present invention will now be described, by way of example, with reference to the accompanying drawings, in which:-

Figure 1 is a block diagram of a general purpose computer;

Figure 2 is a block diagram of software components embodied in a computer employing resource scheduling according to the present invention;

Figure 3 illustrates part of a CPU time reservation method according to the present invention;

Figure 4 is a flow chart illustrating the operation of the CPU secondary scheduler of Figure 2; and

Figure 5 is a flow chart illustrating an alternative manner of operation of the CPU scheduler of Figure 2.

Referring to Figure 1, a general purpose computer comprises a CPU 1, RAM 2, ROM 3, a video interface card 4, a hard disk drive controller 5, a keyboard interface 6 and a mouse interface 7 all interconnected by a combined data and address bus 8. A video monitor 9 is connected to the output of the video interface card 4. A hard disk drive 10 is connected to the hard disk drive controller 5. A keyboard 11 is connected to the input of the keyboard interface 6 and a mouse 12 is connected to the input of the mouse interface 7.

The general purpose computer is operating under the control of a multi-tasking operating system. The operating system may be a known

operating system such as Windows NT, Unix or Linux but with the resource scheduling modified to operate as described below.

Referring to Figure 2, in order to provide resource sharing the operating system comprises a dispatcher component 11, a primary scheduler component 12, a CPU reservation component 13, a hard disk drive reservation component 14, a memory reservation component 15, a CPU secondary scheduler 16, a hard disk drive secondary scheduler 17 and a memory balance manager 19.

The dispatcher component 11 maintains queues of threads awaiting servicing by the CPU 1. A thread may have a priority value between 0 and 31, 31 being the highest priority, and the dispatcher component 11 maintains a separate queue for each priority. Thus, all the threads with priority 0 are in one queue, all the threads with priority 1 are in another queue and so on. Within each queue, new threads are added to the tail of the queue which operates on a FIFO principle.

The CPU secondary scheduler 16 is responsible for granting access to the CPU 1 to threads in a manner that will be described in more detail below. The hard disk drive secondary scheduler 17 runs a lottery to determine access to the hard disk drive 10.

The primary scheduler 12 translates high-level abstract resource reservation requests, made by or on behalf of application components 22, into a form suitable for use by the reservation components 13, 14, 15. The reservation components 13, 14, 15 attempt to reserve the requested resources.

A guaranteed service application 21 for use with the present embodiment, consists of a plurality of a components 22 that are instantiated as necessary. Each component 22 comprises at least one thread 23. The constructor 24 of each component 22 includes a procedure which calls CPU, hard disk drive and memory reservation methods of the primary scheduler 12. The component 22 indicates its level of need for a resource by passing the application's process ID (which is unique to the application 21), an indication of the amount of resource required as parameters when calling the reservation methods and optionally a duration for the reservation. Alternatively, the application 21 itself or a management entity, can make a

reservation on behalf of the guaranteed service application 21. A best effort service application 25 consists of a plurality of a components 26 that are instantiated as necessary. The best effort service application's components 26 do not make resource reservations or have reservations made for them.

Considering now the case of a request for CPU time, when the CPU reservation method of the primary scheduler 12 is invoked by an application component's constructor 24, the CPU reservation method maps a percentage of resource time, received as a reservation parameter, into a component "signature". The component signature comprises a one-dimensional array of up to 256 elements, each element of which is a record comprising fields for the application's process ID for the component 22, a reservation ID, the forwards flexibility and the backwards flexibility. Initially, the fields of the elements of the array are empty. The fields of selected elements of the array are then filled using the parameters of the call made by the component 22. For instance, if the percentage value in the call is 10%, every tenth element of the array is filled and if the percentage value us 20%, every fifth element of the array is filled in. This mapping generally aims to achieve the finest grained reservation signature possible.

Once the component signature has been completed, the CPU reservation component 13 is instantiated on demand by the primary scheduler 12. When the CPU reservation component 13 is instantiated, the primary scheduler 12 forwards the component signature, application and reservation Ids and any duration to the CPU reservation component 13 via a method invocation.

The CPU reservation component 13 then tries to merge the component signature into a copy 31 of a master signature 32. The component signature is replicated to fill the full lengths of the signature arrays 31, 32 used by the CPU reservation component 13 and the CPU secondary scheduler 16. For example a reservation signature of 25% maps onto a 1 in 4 slot signature which is repeated 64 times. If any elements of the component signature clashes with an existing reservations (see Figure 3), the CPU reservation component 13 tries moving the clashing elements of the component signature according to their forwards and backwards flexibility values and moving the clashing elements of the copy 31 of the

master signature 32, according to their backwards and forwards flexibilities, until the component signature fits into the copy 31 of the master signature 32. If this cannot be achieved an exception is raised and communicated back to the component 22 via the primary scheduler 12. If merging is achieved, the master signature 32 is updated according to the modified copy 31 and this information is communicated back to the primary scheduler 12. The primary scheduler 12 then returns to reservation ID to the calling component 22.

The CPU reservation component 13 always maintains at least a predetermined minimum number of elements of the master signature 32 free for the set of best effort applications 25. Each element of the master signature 32 corresponds to one CPU access time slice, e.g. 20ms. The CPU reservation component 13 automatically deletes reservations, requested with a duration parameter, that have time-expired.

The CPU secondary scheduler 16 uses the master signature 32 cyclically to allocate CPU time to components 22. The use of allocated CPU time by the threads 23 within a component 22 is the responsibility of the application programmer.

Priority based scheduling is used to schedule threads 23 which are very high priority threads (above priority level 15) that are essential to the integrity of the system. This means that threads 23 associated with time-critical operating system tasks, do not have to undergo the process of reservation and admission, as described above. The master signature 32 is used specifically for priority levels 0-15. The reason for this approach is to make possible support for legacy operating system code and services that are essential to the stability of the system.

The CPU secondary scheduler 16, uses the standard Round-Robin scheduling algorithm for threads with a priority greater than 15. Generally, these threads will not require a complete time-slice and are therefore negligible in terms of CPU access time.

Referring to Figure 4, the CPU secondary scheduler 16 repeatedly performs the following process. The CPU secondary scheduler 16 determines whether a new thread is pending in the dispatcher 11 (step s1). If a new thread is pending, the CPU secondary scheduler 16 determines whether a thread is currently running (step s2). If a thread is currently running, it determines whether the new pending

thread has a higher priority than the running thread (in this context a guaranteed service thread has "priority" over a best-effort service thread with the same priority) (step s3). If the new thread's priority is higher, the CPU secondary scheduler 16 pre-empts the running thread, dispatches the new thread and places the pre-empted thread at the head of its priority level queue in the dispatcher 11 (step s4).

If, at step s1, a new thread is not ready or, at step s2, no thread is running, the CPU secondary scheduler 16 determines whether the current time slice has expired (step s5). If the current time-slice has not expired then the process returns to step s1 else dispatches the thread (step s7) at the head of the dispatcher queue for the highest priority greater than 15 (step s6).

If, at step s6, no threads with priorities greater than 15 are pending, the CPU secondary scheduler 16 inspects the next element of the master signature (step s8). If the element is empty, the CPU secondary scheduler 16 dispatches the thread from the front of the highest priority queue that is not empty (step s9). If the element is not empty, the CPU secondary scheduler 16 looks for a thread belonging to the application, identified in the master signature element, in the highest priority populated queue in the dispatcher 11 (step s10). If one is found then it is dispatched (step s11) else the thread from the front of the queue is dispatched (step s12) and the CPU secondary scheduler 16 attempts to move the reservation (step s13). This dynamic slot shift can only be made to the extent of the next slot reserved by the same guaranteed service application 21. The reservation is returned to its correct position for the next cycle through the master signature and if the master signature is updated by the CPU reservation component 13.

It should be noted that whenever a guaranteed service thread cannot be serviced at steps s8 and s10, the best-effort service application 25 has the possibility of having its threads 26 dispatched.

When a component 22 of the guaranteed service application 21 is destroyed, its destructor 27 it may invoke a reservation cancel method of the primary scheduler 12, passing the reservation ID as a parameter. The primary scheduler 12 then invokes a cancellation method of the CPU reservation component 13 which removes the calling component's slot

reservations from the copy 31 of the master signature 32 and then updates the master signature 32.

Considering now the case of disk access, an IRP (I/O Request Packet) is an encapsulated request, which defines the processing requirements, including the type of operation to be carried out (read or write), the number of bytes to manipulate, the synchrony of the request etc..

The constructor 24 of a component 22 includes a procedure for registering the guaranteed service application 21 for disk access. This process calls a disk access reservation method of the primary scheduler 12 with a demand level indicator, usually a percentile, as a parameter. The primary scheduler 12 then calls a disk access reservation method of the hard disk drive reservation component 14 with the demand level indicator as a parameter. The hard disk drive reservation component 14 then allocates a number of "lottery tickets" to the application 21 containing the component 22 in dependence on the demand level indicator. The number of lottery tickets issued may be increased as the demand level indicator is increased. The component 22 may request an abstract reservation in terms of a class identifier which the primary scheduler 12 can use to reference a persistent lookup table mapping the appropriate demand level indicator.

When a thread 23, belonging to a guaranteed service application 21, requires disk access, it sends an IRP to an operating systems I/O manager 20, which is then forwarded to the hard disk secondary scheduler 17 via the operating system's file system component 30. The operating system's file system component 30 increases the granularity of the access request, e.g. from a request for one large block a data to many small blocks of data. When the IRP reaches the hard disk secondary scheduler 17, it is place in a wait array according to the guaranteed service application's process ID. The hard disk drive secondary scheduler 17 runs a lottery by randomly generating "ticket" numbers. If winning application 21 has an IRP in the wait array, the winning application 21 is granted disk access and the longest waiting IRP for the winning application is passed on to a physical disk device driver process 32. When there is no winning ticket holder or the winning ticket holder does not have an IRP ready for servicing, then an alternative IRP is scheduled via a simple cyclic selection scheme which includes IRPs from the best effort service application 25.

When a component 22 of the guaranteed service application 21 is destroyed, its destructor 27 determines whether it is the only instance of a component 22 of the application 21. If it is the last such instance, it invokes a cancel hard disk access method of the primary scheduler 12 which in turn invokes a cancellation method of the hard disk drive reservation component 14. The hard disk drive reservation component 14 then de-allocates the tickets allocated to the application 21.

Considering now the case of memory access, each application 21, 25 maintains its own unique virtual address space, which is a set of memory addresses available for its threads to use. This address space is much larger than the RAM 2 of the computer. When a component 22, 26 references its application's virtual memory, it does so with a pointer which is used by a virtual memory manager to determine the location in RAM 2 to which the virtual address maps. However, some portions of memory may actually reside on disk 10 or backing store. This means that data or code which is infrequently accessed is held on disk 10 thus retaining RAM 2 for better use. To facilitate this scheme, memory is managed in fixed size blocks known as "pages", which are usually 4Kb or 8Kb. If a component 22 references an address whose page is not in RAM 2, then a page fault occurs. This triggers a process known as "paging" which is the task of locating the faulted page within the page file and then loading the page into RAM 2. If there is insufficient space in RAM 2, then a page must be first swapped out and written to the page file. However, each process usually keeps a minimum number of pages locked into RAM 2 known as the "working set".

All modern operating systems generally protect memory in three forms:- physical hardware disallows any thread from accessing the virtual address space of other components, a distinction is made between the mode of operation, kernel-mode (ring 0) which allows threads access to system code and data, and user-mode (ring 3) which does not, and a page-based protection mechanism wherein each virtual page maintains a set of flags that determine the type of access permitted. However, these mechanisms are aimed at protecting a component's memory resources from unwanted interference by other unauthorised processes in the system. They do not prevent a process from consuming more than its reasonable share of memory.

To aid understanding of the approach of the present invention to the implementation of working set size control, the working set management scheme in Windows NT will be briefly described. When created, each component is assigned two thresholds, a minimum working-set size and maximum working-set size. The minimum defines the smallest number of pages the virtual memory manager attempts to keep locked concurrently in physical memory, whilst the maximum defines an expansion range which can be used if the component is causing a considerable number of page faults. If the working set is too small, then the component incurs a large number of paging operations and thus a substantial overhead is incurred through continuously swapping pages to and from the backing store. Alternatively, if the working set is too large, few page faults occur but the physical memory may be holding code and/or data which is infrequently referenced and thus overall efficiency is reduced. In Windows NT, the Memory Manager (MM) adjusts the working sets once every second, in response to page-in operations or when free memory drops to below a given threshold. If free memory is plentiful, the MM removes infrequently referenced pages only from working sets of components whose current size is above a given minimum, this is known as aggressive trimming. However, if free memory is scarce, the MM can force trimming of pages from any component until it creates an adequate number of pages, even beyond the minimum threshold. Of course, components which have extended working sets are trimmed in preference to those which have the minimum working set size.

In the present embodiment, the constructor 24 of a component 22 includes a procedure for reserving a working set. This procedure calls a memory reservation method of the primary scheduler 12 which, in turn calls a method of the memory reservation component 15. The memory reservation component 15 translates this request into the appropriate expansion of a default minimum working set size. The minimum working set size regulates the number of pages an application 21 can concurrently lock (or pin) into RAM 2. Thus, a component 22 that wishes to reserve a portion of memory, makes the reservation request via the primary scheduler 12 to the memory reservation component 15 which then increases the working set thresholds accordingly. This adjustment is facilitated through operating system calls. It is then the responsibility of the component to pin the pages into RAM 2 as required. The pinning process is arbitrated on a first-

come first-serve basis. If insufficient physical pag s are available, the operating system will reject the pin request, therefore the component 22 is exp cted to pin its reserved pages soon after allocation.

If there are insufficient free physical memory resources, the memory reservation component 15 then attempts to force the processes of best-effort service applications 25 to write pages back to file store until sufficient free physical pages are available. If insufficient space can be made, then the reservation is rejected. As with the other resource modules, a portion of the RAM 2 is reserved for a minimal best-effort service. This is facilitated by maintaining a count of the total number of physical pages being used under the guaranteed service and checking this against a portion of the total physical page capacity. It should also be noted that the 'best-effort' service is strictly a minimal guaranteed service (defined by the system assigned minimum working set size) in addition to the best-effort service through potential expansion of the working set. It is assumed that processes cannot bypass the primary scheduler 12 and directly alter their own working set size. Finally, special consideration is made for pages which are being used for shared memory purposes. If any client of a shared page has undergone reservation, then the page is classified as reserved even though other clients may not have made a reservation.

A second embodiment is the same as the first embodiment described above except for the manner of operation of the CPU secondary scheduler 16. The operation of the CPU scheduler of the second embodiment will now be described.

Referring to Figures 2 and 4, the CPU secondary scheduler 16 repeatedly performs the following process. The CPU secondary scheduler 16 determines whether a new thread is pending in the dispatcher 11 (step s101). If a new thread is pending, the CPU secondary scheduler 16 determines whether a thread is currently running (step s102). If a thread is currently running, it determines whether the new pending thread has a higher priority than the running thread (in this context a guaranteed service thread has "priority" over a best-effort service thread with the same priority) (step s103). If the new thread's priority is higher, the CPU secondary scheduler 16 pre-empts the running thread, dispatches the new thread and places the pre-empted thread at the head of its priority level queue in the dispatcher 11 (step s104).

If, at step s101, a new thr ad is not r ady or, at step s102, no thread is running, the CPU secondary scheduler 16 determines whether the current time slice has expired (step s105). If the current time-slice has not expired then the process returns to step s101 else dispatches the thread (step s107) at the head of the dispatcher queue for the highest priority greater than 15 (step s106).

If, at step s106, no threads with priorities greater than 15 are pending, the CPU secondary scheduler 16 inspects the next element of the master signature (step s108). If the element is empty, the CPU secondary scheduler 16 dispatches the thread from the front of the highest priority queue that is not empty (step s109). If the element is not empty, the CPU secondary scheduler 16 looks for a thread belonging to the application, identified in the master signature element, in a queue in the dispatcher 11 (step s110), searching from the priority 15 queue towards the lowest priority queue and from the head of each queue to the tail thereof. If one is found then it is dispatched (step s111) else the thread from the front of the highest priority populated queue is dispatched (step s112) and the CPU secondary scheduler 16 attempts to move the reservation (step s113). This dynamic slot shift can only be made to the extent of the next slot reserved by the same guaranteed service application 21. The reservation is returned to its correct position for the next cycle through the master signature and if the master signature is updated by the CPU reservation component 13.

It should be noted that whenever a guaranteed service thread cannot be serviced at steps s108 and s110, the best-effort service application 25 has the possibility of having its threads 26 dispatched.

The present invention has been described with reference to applications comprising components which in turn comprise one or more threads. It will be appreciated that the present invention is not limited to this situation and may be implemented in a system in which application programs are not built using object-oriented languages.

Whilst the present invention has been described with reference to CPU, hard disk and memory access, it will be appreciated that it may be applied to the scheduling of access to other resources.

It will be appreciated that the term "lottery ticket" is used figuratively.

CLAIMS

1. A method of administering resource utilisation in a computer, the method comprising the steps of:

running a first process to make a reservation for access to a resource in dependence on a resource requirement communication from an application process;

running a second process to grant requests for access to said resource from said application process in dependence on said reservation;

creating a scheduling means having a method or methods for processing reservation requests for a plurality of resources and initiating resource specific reservation processing;

creating a reservation means having a method or methods for making reservations for access to a resource;

said application process calling a method of the scheduling means, said method taking a first resource access requirement definition as a parameter;

said method of the scheduling means calling a reservation method of the reservation means to make a reservation for said application process, the reservation method taking a second resource access requirement definition as a parameter;

running a resource specific scheduling process to grant access to a resource in dependence on the reservation made by the reservation means; and

utilising said resource for the purposes of said application process.

2. A method according to claim 1, wherein said method of the scheduling means translates the first resource requirement definition into the second resource requirement definition.

3. A method according to claim 1 or 2, wherein said resource is a CPU.

4. A method according to claim 1 or 2, wherein said resource is a mass storage device.

5. A method of scheduling access to a CPU, the method comprising the steps of:

generating a one-dimensional reservation request pattern; and

merging the reservation request pattern with a one-dimensional CPU access control pattern, representing empty CPU access time slots and reserved CPU access time slots, without substantially disturbing either the reservation request pattern or the reserved CPU access time slots in the reservation request pattern.

6. A method according to claim 5, wherein said merging step comprises relocating a non-empty time slot element of the reservation request pattern or the CPU access control pattern such that the patterns can be merged without any reserved CPU access time slot elements being deleted or overwritten.

7. A method according to claim 6, wherein the relocated non-empty time slot element is relocated by an amount defined in said time slot element.

8. A method according to claim 1, wherein said first process is a method according to claim 5, 6 or 7.

9. A method of granting access to a CPU comprising:

generating a one-dimensional CPU access control pattern, each element of which relates to a quantum of CPU access time; and

at the end of a quantum of CPU access time:

granting access to any pending processes having a priority greater than a predetermined level; and then

if the next pattern element is empty then granting access to a pending process meeting a predetermined prioritisation criterion else granting access to a process identified in the pattern element.

10. A method according to claim 9, wherein pending processes populate queues having different priorities and access is granted to the process identified in the pattern element when there is not a populated process queue having a higher priority than the queue in which said process is present.

11. A method according to claim 9 or 10, wh rein the one-dimensional CPU access control pattern is generated by a method according to claim 5, 6 or 7.

12. A method according to claim 1, wherein the second process performs a method according to claim 10 or 11.

13. A method of administering resource utilisation in a computer, the method comprising:

running a first process to make a reservation for access to a resource in dependence on a resource requirement communication from an application process; and

running a second process to grant requests for access to said resource from said application process in dependence on said reservation, wherein

said first process comprises allocating one or more resource tokens to said application process in dependence on the second resource access requirement definition and

the second process performs a method comprising the steps of:

(i) storing requests for access to a mass storage device from application processes;

(ii) generating substantially randomly a resource token identifier; and

(iii) if no application process has been allocated said identified resource token then passing on to a mass storage device driver process the stored request for access from an application process selected on the basis of a predetermined prioritisation criterion else passing on to a mass storage device driver process a stored request for access from an application process to which said identified resource token was allocated.

14. A method of administering resource utilisation in a computer, the method comprising:

running a first process to make a reservation for access to a resource in dependence on a resource requirement communication from an application process; and

running a second process to grant requests for access to said resource from said application process in dependence on said reservation, wherein

the first process determin s a w ighting function associated with the application process;

the second process performs a method comprising the steps of:

(i) storing requests for access to a mass storage device from application processes;

(ii) using a stochastic process, either selecting an application process with a probability determined by the weighting associated with the application process and passing on to a mass storage device driver process the stored request for access from the selected application process, or passing on to a mass storage device driver process a stored request for access from an application process selected on the basis of a predetermined prioritisation criterion

15. A computer configured to operate in accordance with a method according to any preceding claim.

16. A data carrier containing computer code for loading into a computer for the performance of the method of any of claims 1 to 14.

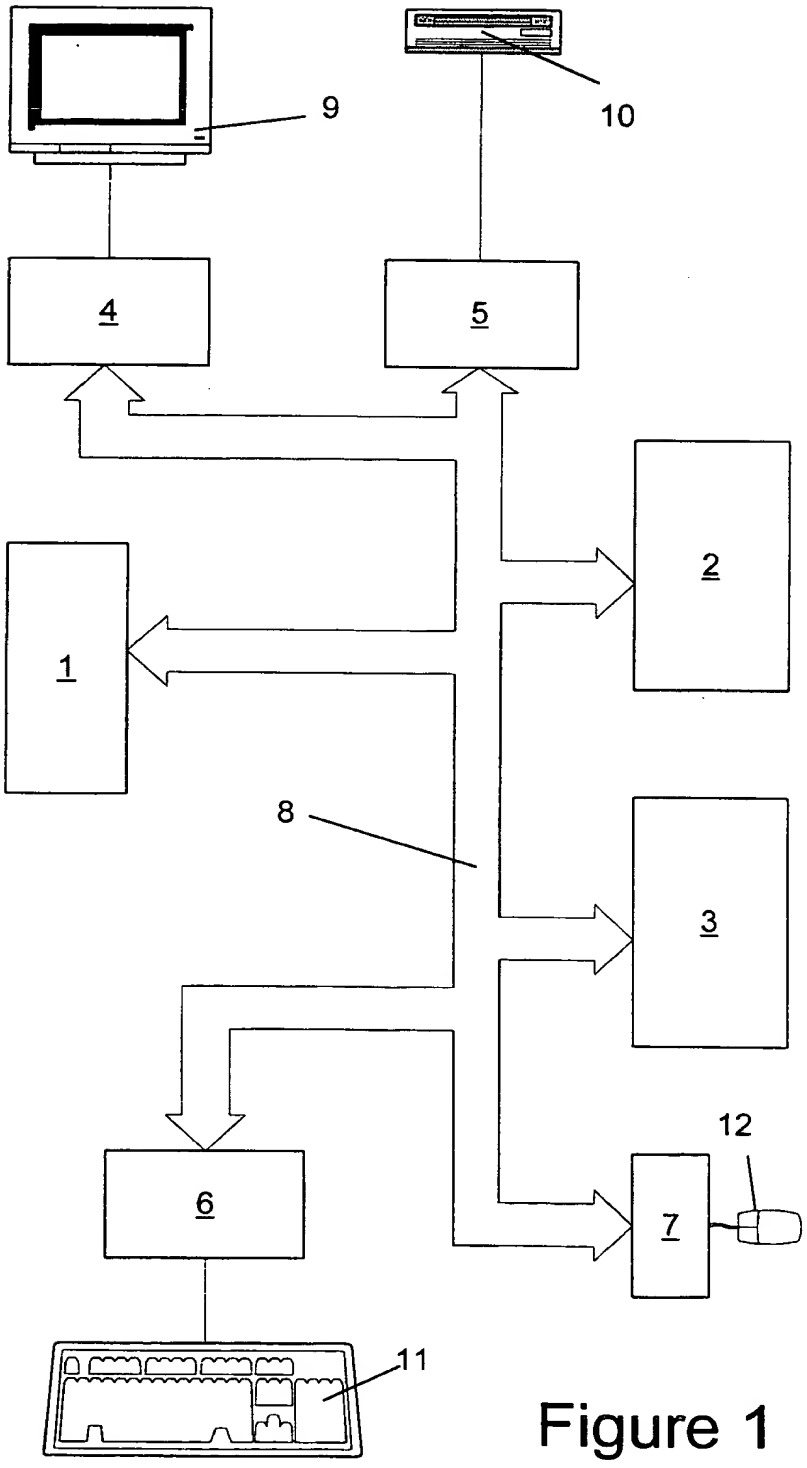


Figure 1

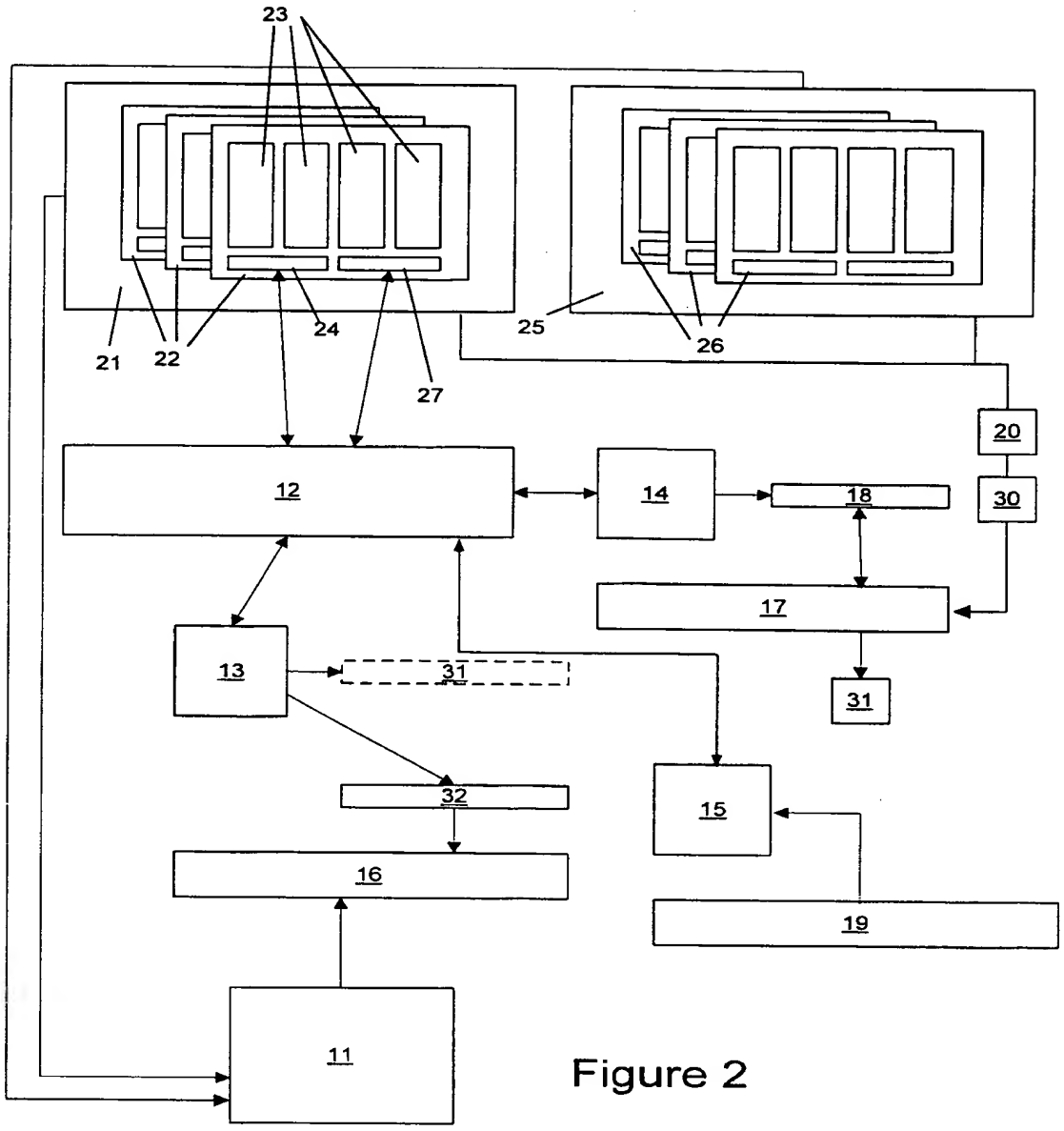


Figure 2

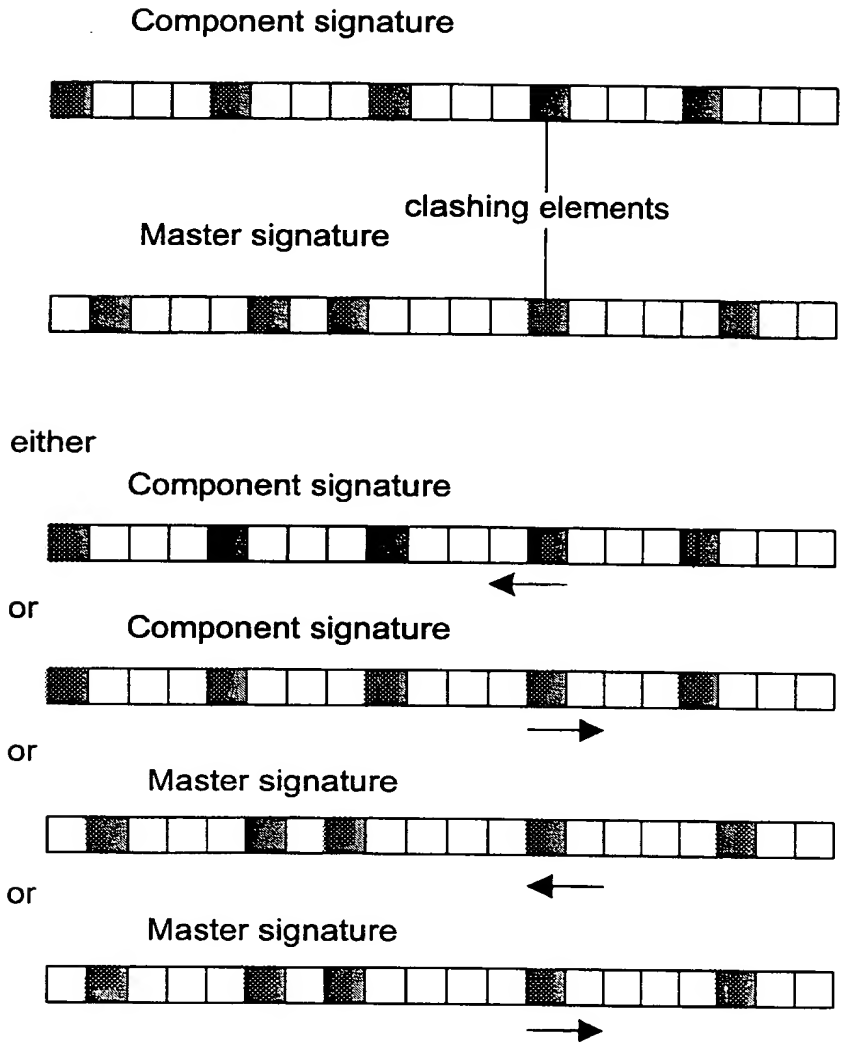

Component signature
clashing elements
Master signature
either
Component signature
or
Component signature
or
Master signature
or
Master signature

Figure 3

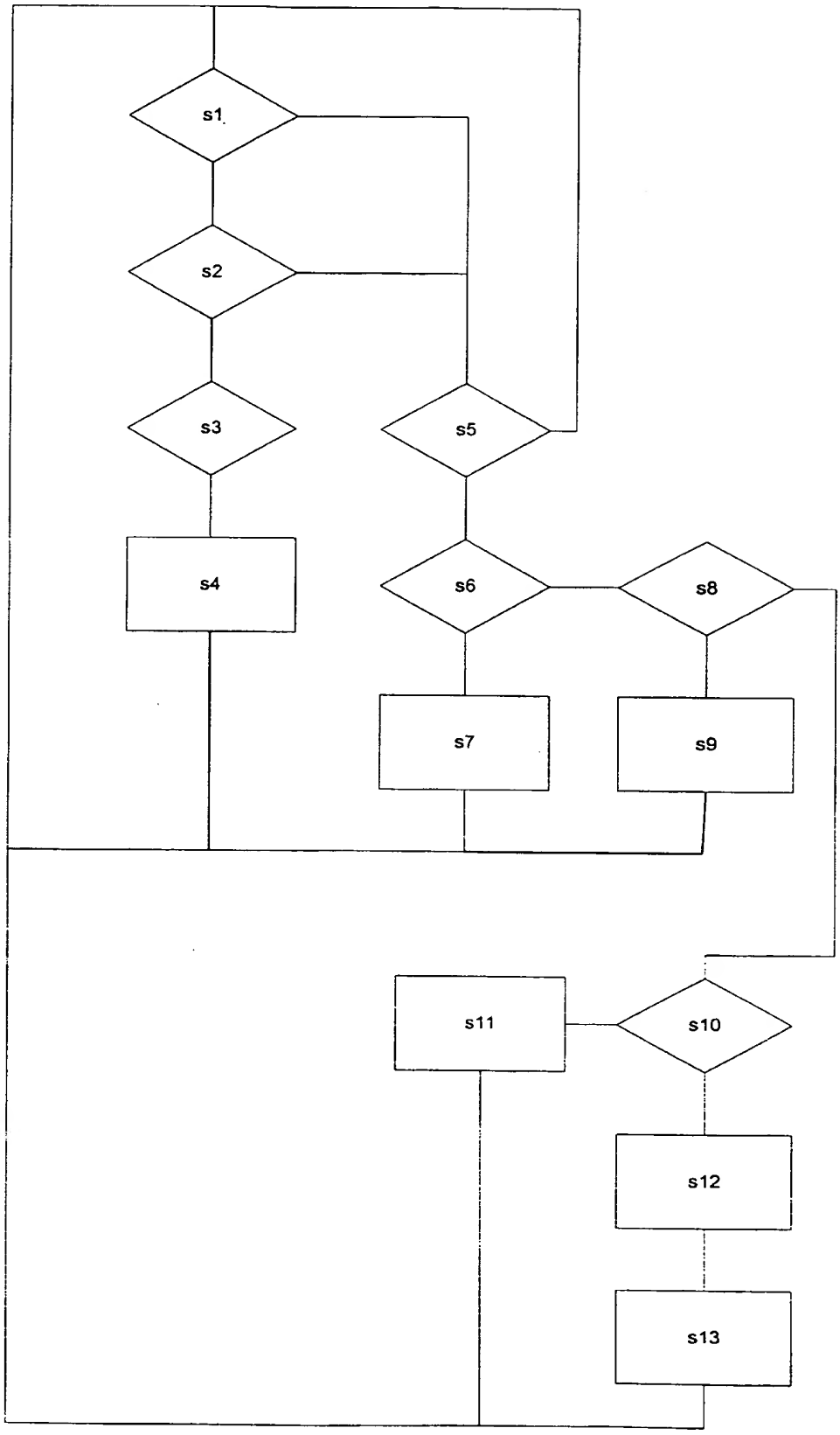


Figure 4

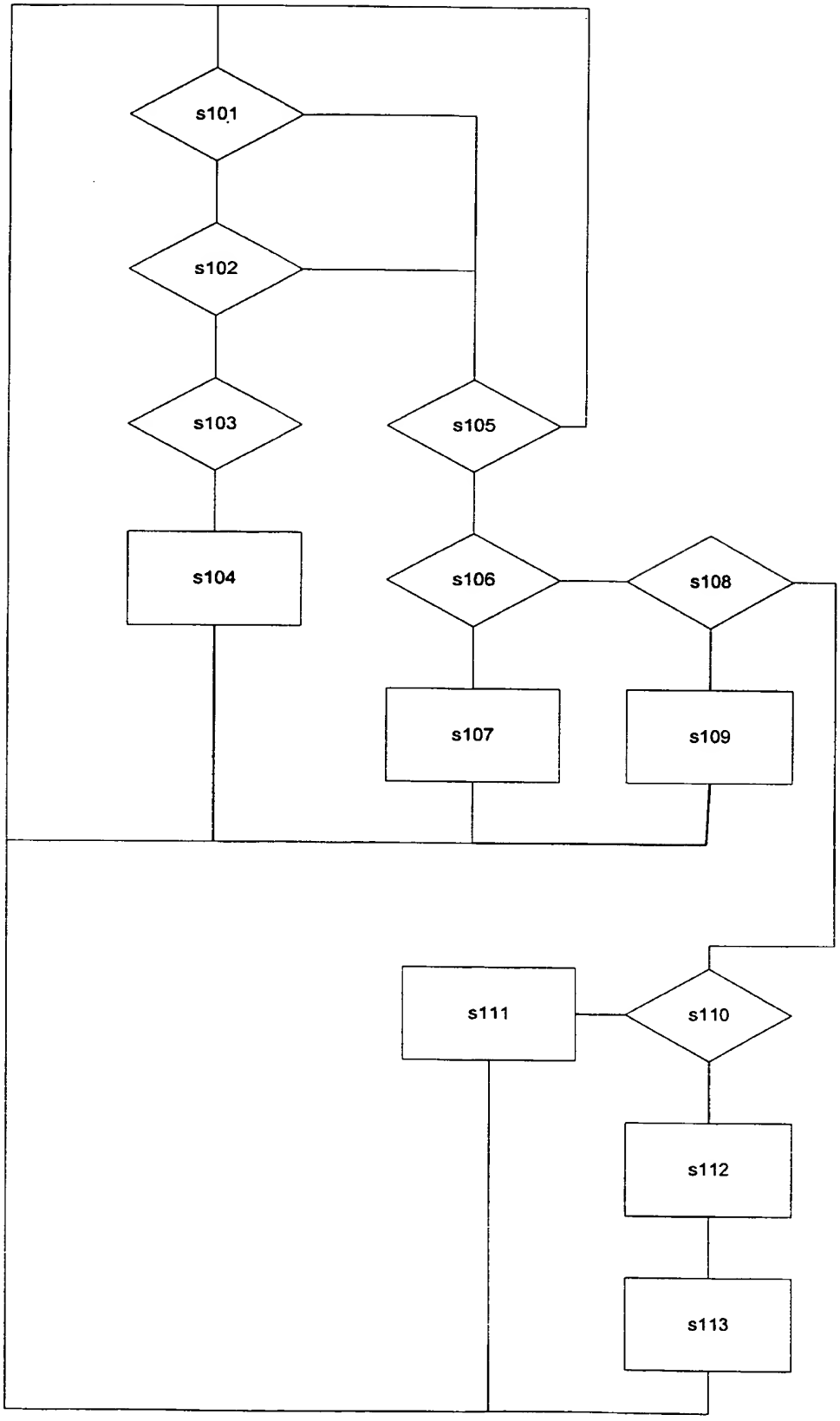

s101
s102
s103
s104
s105
s106
s107
s108
s109
s110
s111
s112
s113

Figure 5

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
21 September 2000 (21.09.2000)

PCT

(10) International Publication Number
WO 00/55732 A3

(51) International Patent Classification[7]: G06F 9/48, 9/50

(21) International Application Number: PCT/GB00/00783

(22) International Filing Date: 3 March 2000 (03.03.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
99301952.0 15 March 1999 (15.03.1999) EP
99303936.1 20 May 1999 (20.05.1999) EP

(71) Applicant *(for all designated States except US)*: **BRITISH TELECOMMUNICATIONS PUBLIC LIMITED COMPANY** [GB/GB]; 81 Newgate Street, London, EC1A 7AJ (GB).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **WADDINGTON, Daniel, Giles** [GB/GB]; Flat 1, 25 St. Georges Quay, Lancaster, Lancashire LA1 4YR (GB).

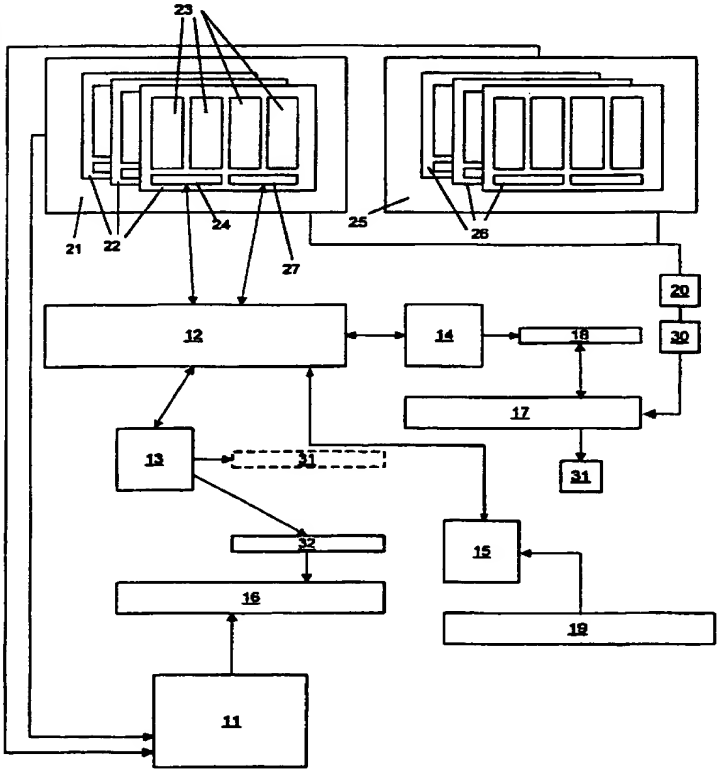
(74) Agent: **WILSON, Peter, David**; BT Group Legal Services, Intellectual Property Department, Holborn Centre, 8th floor, 120 Holborn, London EC1N 2TE (GB).

(81) Designated States *(national)*: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent

[Continued on next page]

(54) Title: RESOURCE SCHEDULING



(57) Abstract: A method of administering resource utilisation in a computer comprises running a first process (13, 14, 15) to make a reservation for access to a resource in dependence on a resource requirement communication from an application (21) and running a second process (16, 17, 19) to grant requests for access to said resource from said application in dependence on said reservation. A further process (12) provides a common interface between the first process (13, 14, 15) for each resource and the application (21). The further process (12) converts high-level abstract resource requirement definitions into formats applicable to the first process (13, 14, 15) for the resource in question. The processes are preferably implemented at methods of software objects.

WO 00/55732 A3

(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*

(88) Date of publicati n of the international search report:
28 December 2000

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

INTERNATIONAL SEARCH REPORT

International Application No
T/GB 00/00783

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F9/48 G06F9/50

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 0 798 638 A (HITACHI LTD) 1 October 1997 (1997-10-01) column 8, line 22 -column 14, line 20 column 32, line 56 -column 33, line 24 ---	1-6,8,9, 11,12, 15,16
A	US 5 812 844 A (DRAVES JR RICHARD P ET AL) 22 September 1998 (1998-09-22) column 6, line 56 -column 10, line 23 claims 1-3 ---	1-9,11, 12,15,16
A	EP 0 817 041 A (SUN MICROSYSTEMS INC) 7 January 1998 (1998-01-07) column 2, line 54 -column 5, line 31 claims 1-7 --- -/--	1-4

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
7 September 2000	13/09/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Bijn, K

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

International Application No
PCT/GB 00/00783

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 247 677 A (WELLAND ROBERT V ET AL) 21 September 1993 (1993-09-21) column 5, line 24 -column 6, line 53	1,13
A	EP 0 790 557 A (MATSUSHITA ELECTRIC IND CO LTD) 20 August 1997 (1997-08-20) page 8, line 22 -page 13, line 38	1,13
A	EP 0 658 841 A (IBM) 21 June 1995 (1995-06-21) column 5, line 34 -column 7, line 8	1,9

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
/GB 00/00783

Patent document cited in search report	Publication dat	Patent family member(s)	Publication dat
EP 0798638 A	01-10-1997	CA 2200929 A	28-09-1997
		JP 2904483 B	14-06-1999
		JP 9319597 A	12-12-1997
		US 5944778 A	31-08-1999
US 5812844 A	22-09-1998	NONE	
EP 0817041 A	07-01-1998	US 5826082 A	20-10-1998
		JP 10063519 A	06-03-1998
US 5247677 A	21-09-1993	FR 2691557 A	26-11-1993
		JP 6035726 A	10-02-1994
EP 0790557 A	20-08-1997	JP 9282184 A	31-10-1997
EP 0658841 A	21-06-1995	US 5487170 A	23-01-1996
		JP 7200318 A	04-08-1995